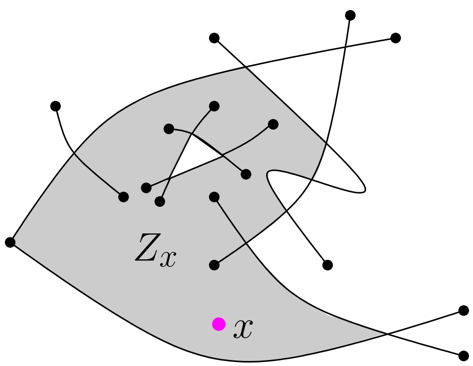
Offline Bewegungsplanung: Zellen eines Konfigurationsraumes

Elmar Langetepe
University of Bonn

Zellen: Th. 2.18

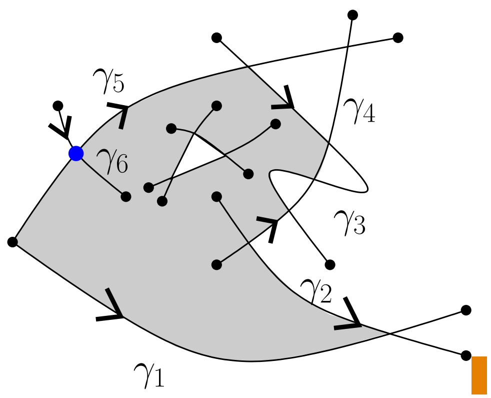
A Arrangement von n Kurvenstücken von denen sich zwei nur s mal schneiden. Jede Zelle von A hat Komplexität $O(\lambda_{s+2}(n))$.

Weniger als $\Omega(n^2)$!! Beweis!!



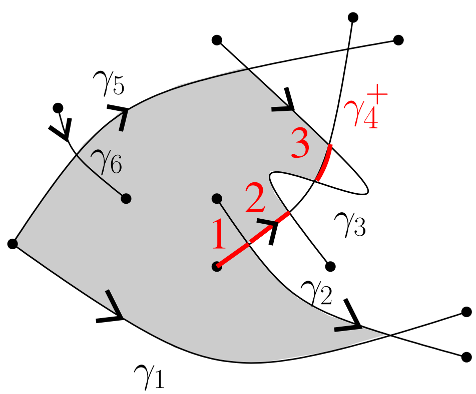
Beweis: Th. 2.18

- Rand zerfällt in mehere Zyklen
- Analyse: Äußerer Zyklus C reicht! $\lambda_s(n)$, n Segmente
- Bezeichnen, orientieren: links nach rechts
- Zyklische Abfolge
  $S = < \gamma_5^- \gamma_1^+ \gamma_2^- \gamma_4^- \gamma_4^+ \gamma_2^- \gamma_2^+ \gamma_4^+ \gamma_3^- \gamma_4^+ \gamma_3^- \gamma_5^- \gamma_6^+ \gamma_6^- >$
- $2n$ Segmente γ_i^-,γ_i^+



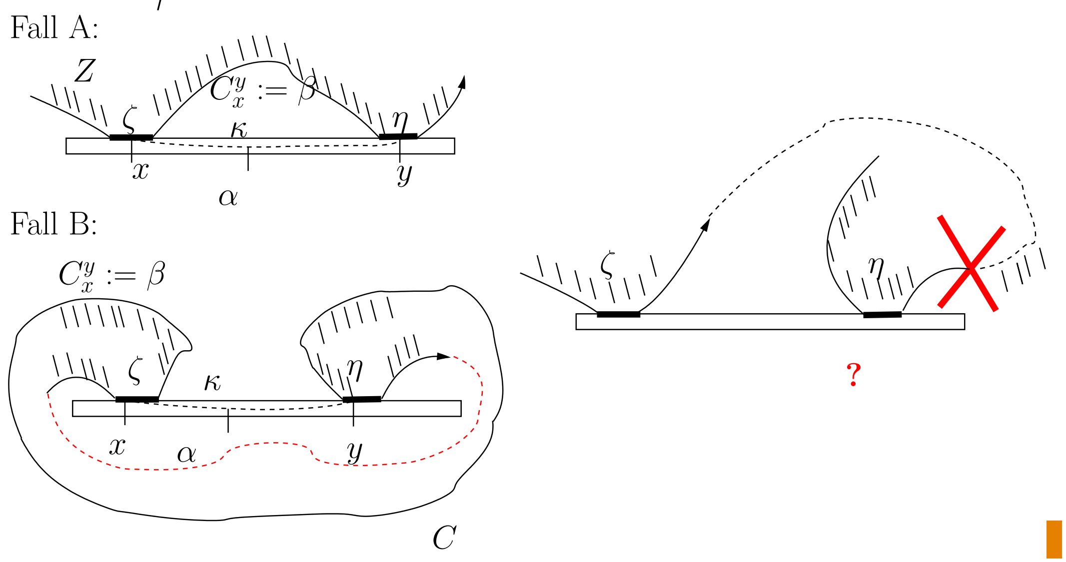
Bogenteile in Reihenfolge! Lem. 2.19

- Zyklische *Folge* des Zykels C:
  $S = < \gamma_5^- \, \gamma_1^+ \, \gamma_2^- \, \gamma_4^- \, \gamma_4^+ \, \gamma_2^- \, \gamma_2^+ \, \gamma_4^+ \, \gamma_3^- \, \gamma_4^+ \, \gamma_3^- \, \gamma_5^- \, \gamma_6^+ \, \gamma_6^- >$
- Bogen γ_i: Segmente von $\gamma_i^+(\gamma_i^-)$ erscheinen in C in derselben Reihenfolge wie entlang von $\gamma_i^+(\gamma_i^-)$
- Beispiel γ_4^+! Beweis!



Beweis: Konsistenzlemma 2.19

- γ_i etwas aufblasen; $\zeta, \eta \in \gamma_i^*$ konsekutive Subsegmente in C!
- Nur zwei Fälle gemäß Innerem; sonst keine Verbindung!!
- Betrachte α und $\beta := C_x^y$
- β berührt γ_i^* nicht: Konsekutiv in C!
- C schnittfrei: $\alpha \cup \beta$ trennt κ von C ab



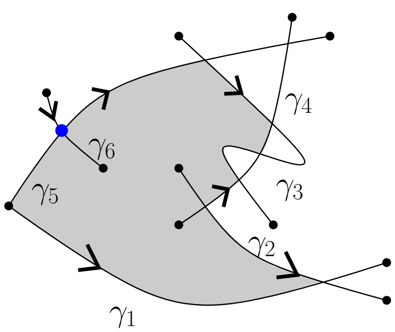
Lineare Sequenz S'' bilden!

- Zyklische Sequenz:

  $S = < \gamma_5^- \, \gamma_1^+ \, \gamma_2^- \, \gamma_4^- \, \gamma_4^+ \, \gamma_2^- \, \gamma_2^+ \, \gamma_4^+ \, \gamma_3^- \, \gamma_4^+ \, \gamma_3^- \, \gamma_5^- \, \gamma_6^+ \, \gamma_6^- >$ auftrennen
- Orientierte Sequenz:

  $S' = \{\gamma_5^- \, \gamma_1^+ \, \gamma_2^- \, \gamma_4^- \, \gamma_4^+ \, \gamma_2^- \, \gamma_2^+ \, \gamma_4^+ \, \gamma_3^- \, \gamma_4^+ \, \gamma_3^- \, \gamma_5^- \, \gamma_6^+ \, \gamma_6^-\}$
- Orientierte Reihenfolge stimmt nicht! Bsp.:γ_5^-
- Verdoppeln: $\gamma_i^- \; (\gamma_i^+) \Rightarrow \gamma_{i,1}^-, \, \gamma_{i,2}^-(\gamma_{i,1}^+, \, \gamma_{i,2}^+)$
- Nun 4n Segmente: Sequenz

  $S'' = \{\gamma_{5,1}^- \, \gamma_1^+ \, \gamma_2^- \, \gamma_4^- \, \gamma_4^+ \, \gamma_2^- \, \gamma_2^+ \, \gamma_4^+ \, \gamma_3^- \, \gamma_4^+ \, \gamma_3^- \, \gamma_{5,2}^- \, \gamma_6^+ \, \gamma_6^-\}$



S'' DSS mit $(4n, s+2)$: Lem. 2.20

- Definition DSS
- Zwischen zwei $\gamma_{i,j}^{-}(\gamma_{i,j}^{+})$ kommt stets anderes Segment vor
- Zz.: Zwei Buchstaben ζ und η wechseln nicht mehr als $s+2$ mal
- Widerspruchsbeweis: Annahme $s+3$ Wechsel!
- Situation:
  $S'' = (\cdots\zeta_1\cdots\eta_1\cdots\zeta_2\cdots\eta_2\cdots\cdots\zeta_j\cdots\eta_j\cdots\zeta_k\cdots(\eta_k\cdots))$

Lem. 2.20

$$S'' = (\cdots \zeta_1 \cdots \eta_1 \cdots \zeta_2 \cdots \eta_2 \cdots \cdots \zeta_j \cdots \eta_j \cdots \zeta_k \cdots (\eta_k \cdots))$$

Fasse je vier zusammen:

$$\begin{array}{llllll} (\zeta_1 & ,\eta_1 & ,\zeta_2 & ,\eta_2) & & \\ & (\eta_1 & ,\zeta_2 & ,\eta_2, & \zeta_3) & \\ & & (\zeta_2 & ,\eta_2, & \zeta_3, & \eta_3) \\ & & & \vdots & & \end{array}$$

Jeweils Schnitt zw. $((\zeta_1, \zeta_2), (\eta_1, \eta_2))$, $((\eta_1, \eta_2), (\zeta_2, \zeta_3))$,...

$s + 3$ ungerade: η_k ex. und $k = \frac{s+2}{2} + 1$ Induktion!!

($(s + 3)$ gerade Übung!)

Lem. 2.20

$s+3$ ungerade: η_k ex. und $k = \frac{s+2}{2} + 1$ Induktion!!

$$
\begin{array}{llllll}
(\zeta_1 & ,\eta_1 & ,\zeta_2 & ,\eta_2) & & \\
 & (\eta_1 & ,\zeta_2 & ,\eta_2, & \zeta_3) & \\
 & & & \vdots & & \\
 & & (\zeta_{\frac{s+2}{2}} & ,\eta_{\frac{s+2}{2}} & ,\zeta_{\frac{s+2}{2}+1} & ,\eta_{\frac{s+2}{2}+1})
\end{array}
$$

ζ führt $\frac{s+2}{2}$ Quadrupel an! η führt $\frac{s+2}{2} - 1$ Quadrupel an!

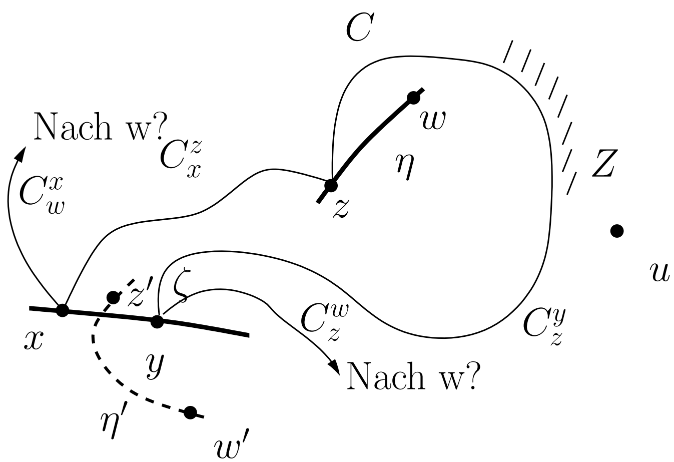
Insgesamt: $2 \cdot (\frac{s+2}{2}) - 1 = s + 1$ Quadrupel

Noch zu zeigen: Jedes Quadrupel erzeugt Schnitt!

Lem. 2.20

Zu zeigen: O.B.d.A: $(\zeta_i, \eta_i, \zeta_{i+1}, \eta_{i+1})$ erzeugt einzelnen Schnitt zw. (ζ_i, ζ_{i+1}) und (η_i, η_{i+1})

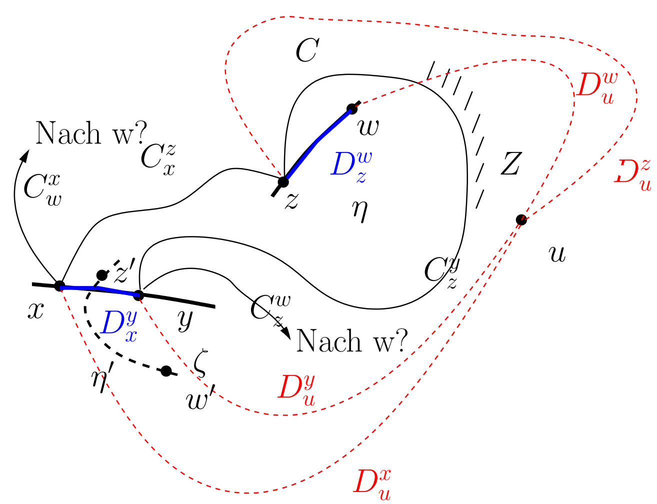
Situation wie folgt:



Zeige, dass Schnitt existieren muss!

Situation: u im Innern!

- u *sieht* x, z, y, w
- Verb. D_u^x, D_u^z, D_u^y, D_u^w schnittfrei mit C_x^z, C_z^y, C_y^w, C_w^x
- Annahme: D_x^y und D_z^w schnittfrei $\Rightarrow$ alle schnittfrei!!
- Entspricht: $K5$ in der Ebene schnittfrei realisiert! Widerspruch!!



Konklusion: Lem. 2.20

- n Bögen, je s Schnitte: Zeige: DSS mit $(4n, s+2)$
- Annahme: DSS mit mind. $(4n, s+3)$
- $(s+3)$ Wechsel auf ζ und η, sukzessive
- Bei jedem Quadrupel $(\zeta_i, \eta_i, \zeta_{i+1}, \eta_{i+1})$ (oder $(\eta_i, \zeta_{i+1}, \eta_{i+1}, \zeta_{i+2})$: Schnitt zwischen (ζ_i, ζ_{i+1}) und (η_i, η_{i+1})
- $s+1$ Quadrupel $\Rightarrow$ $s+1$ Schnitte, Widerspruch!!
- DSS mit $(4n, s+2)$

Zurück zur Aufgabe: Konfigurationsraum

- n Bögen
- Je zwei schneiden sich s mal
- X-monoton, eventuell erzeugen
- Startpunkt x
- Komplexität der Zelle Z_x: $\lambda_{s+2}(4n)$
- Divide and Conquer Ansatz sinnvoll

Alg. 2.4: Zelle Z_x!

Gegeben: Punkt x, n X-monotone Bögen.

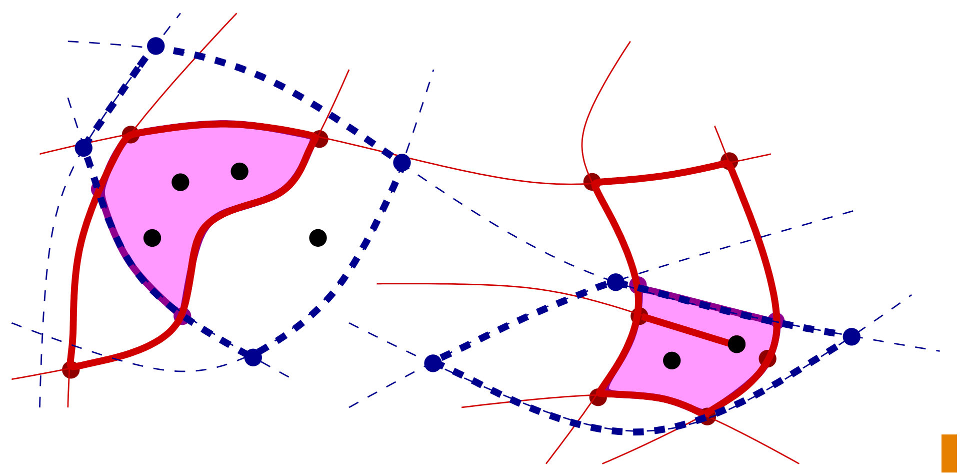
Gesucht: Zelle Z_x im Arrangement der Bögen, die x enthält.

- Zerlege Menge der Bögen in gleichgroße Teilmengen R und B.
- Berechne rek. im Arrangement $A(R)$ Zelle $Z(R)_x$, die x enthält.
- Berechne rek. im Arrangement $A(B)$ Zelle $Z(B)_x$, die x enthält.
- Berechne Zusammenhangskomponente Z_x von $Z(R)_x \cap Z(B)_x$, die x enthält und berichte diese! RED-BLUE Merge

Zuerst RED-BLUE Merge betrachten, dann zurück!

Allgemeiner RED-BLUE Merge

- Rotes Arrangement R mit Zellen $R_1, \ldots, R_{m_R}$, r Ecken.
- Blaues Arrangement B mit Zellen $B_1, \ldots, B_{m_B}$, b Ecken.
- Punktmenge p_i $i = 1, \ldots, k$
- Schnittzellen $Z_j = R_{\mu_j} \cap B_{\nu_j}$, $j = 1, \ldots, l$, die mind. ein p_i enthalten



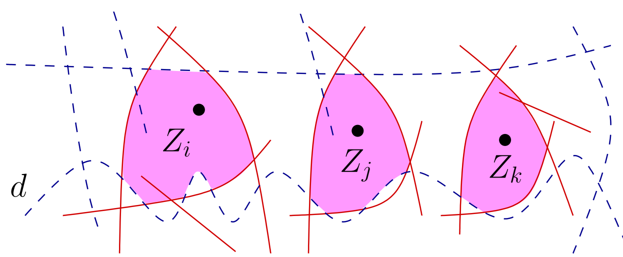
Komplexität der Schnittzellen: Lem. 2.21

Kombinationslemma: Guibas,Sharir,Sifrony 1989 (DSS *Bibel*)

Komplexität der Zellen $Z_1, \ldots, Z_\ell$, $\ell \leq k$:

$$|Z_1| + |Z_2| + \cdots + |Z_\ell| \in O(r + b + k)$$

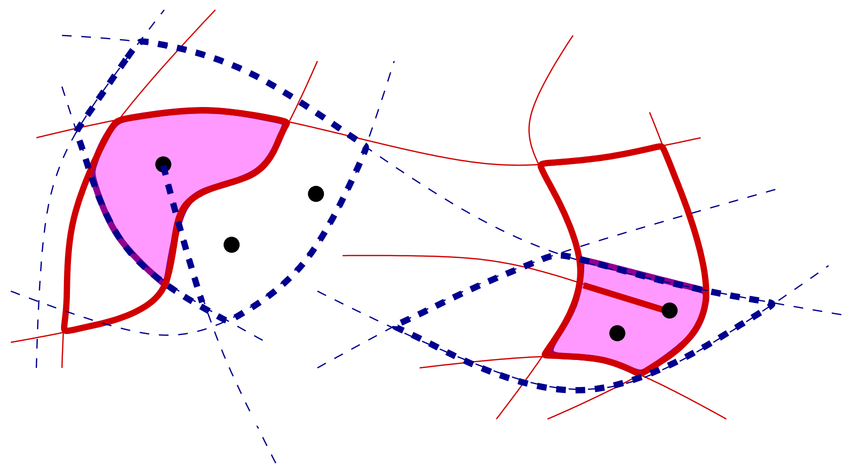
Nicht trivial:



Zur Analyse der Berechnung verwenden!!!

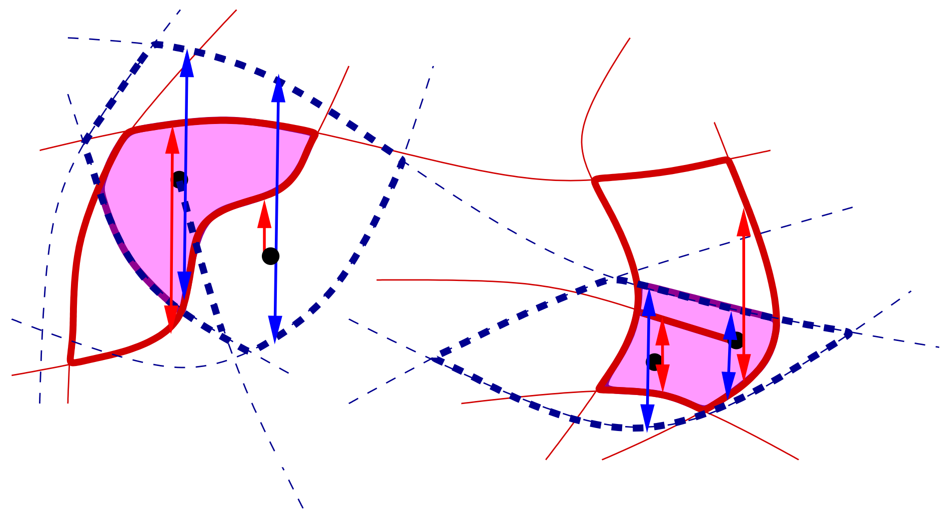
Berechnung: Th. 2.22

- Rotes Arrangement R (r), Blaues Arrangement B (b), Punktmenge P (k), Schnittzellen Z_i
- $|Z_1| + |Z_2| + \cdots + |Z_\ell|$ in $O((r+b+k)\log(r+b+k))$ berechnen!
- Sweep Algorithmus
- P erweitern: *Innere* Endknoten von R und B: Wichtig!!
- Q: P und *innere* Endknoten



Alg. 2.5: Preprocessing!

Für alle $q \in Q$ darüber/darunter-liegende Kante in R und B



Durch Sweep in jedem Arrangement: Übung
$O((r + b + k) \log(r + b + k))$

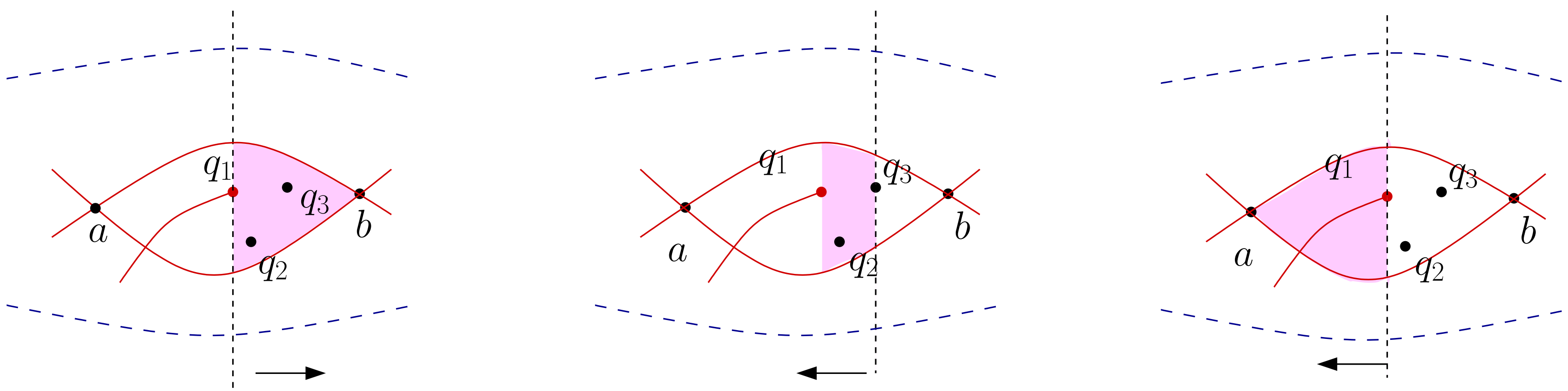
Alg. 2.5: Sweep in zwei Richtungen!

i) Teile der Ergebniszellen: Rechts vom am weitesten links liegenden $q \in Q$ beginnen

ii) Teile der Ergebniszellen: Links vom am weitesten rechts liegenden $q \in Q$ beginnen

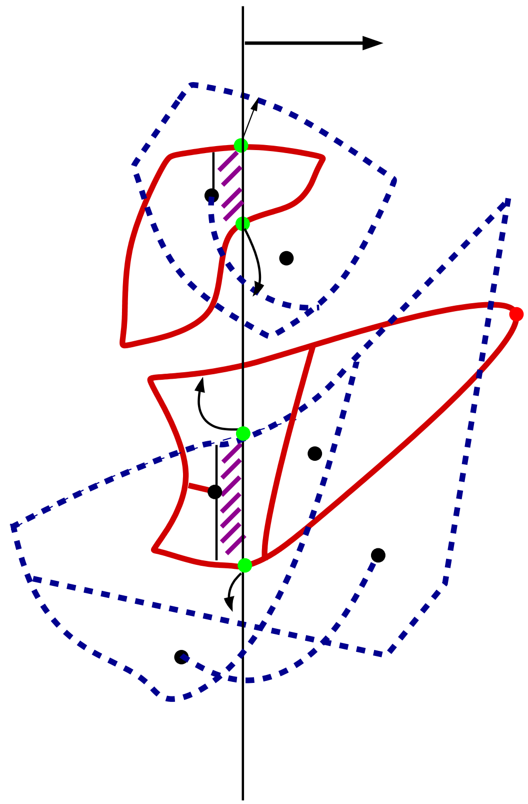
iii) Nochmals Aufteilen in Teilzellen

Dann Vereinigung!



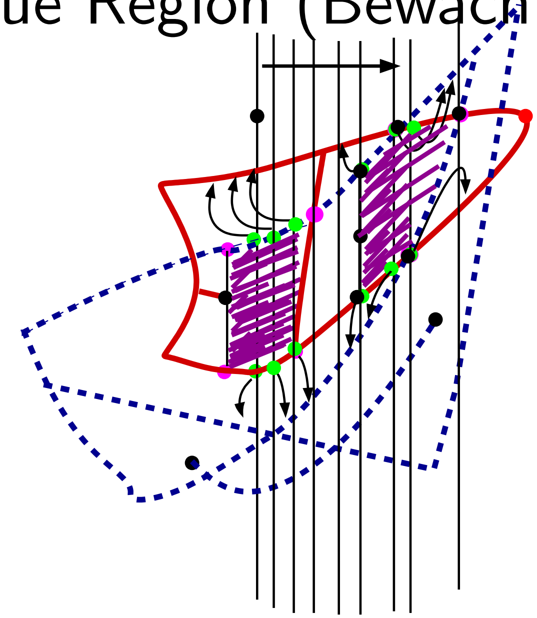
Alg. 2.6: Sweep (eine Richtung!)

- ES: nach X-Koord. sort. Punkte aus Q + zusätzl. Ecken von R und B.
- SSS: Zu jedem Zeitpunkt:
  - sortierte Folge der Schnittzellen entlang der Sweepline
  - Schnittzellen haben Zeiger auf O/U Kanten
  - Scout läuft auf Schnittkante und bewacht mitentscheidene Kanten!!



Alg. 2.6: Ereignisse!!

- Bewachte Kante wechselt! (Schnitte: Mit Bewacher?)
- Randkante wechselt! (Schnitte: Mit Bewacher/Mit Randkante?)
- Red/Blue Schnittpunkt: Region Ende oder Wechsel Rand/Bewachte Kante
- Endpunkt aus Q: Neue Region (Bewachung/Schnitte)!!



Beispiel (Tafel): Alg. 2.6: Ereignisse!!

$n = r + b + k$

1. Rot/Roter Schnittpunkt: Wechsel! Schnitte! $O(1)$
2. Blau/Blauer Schnittpunkt: Wechsel! Schnitte! $O(1)$
3. Rot/Blauer Schnittpunkt: Wechsel! Schnitte! $O(1)$
4. Neuer Punkt: Regionstart $O(1)$ Preprocessing! Einfügen in SSS: $O(\log n)$; Schnitte! $O(1)$

- Schnitte Blau/Rot berechnen: 1), 2), 3), 4)!
  - A) Nächsten berechnet in $O(1)$
  - B) Einfügen in ES: $O(\log n)$: Begründung!

Analyse: Red-Blue Merge Th. 2.22

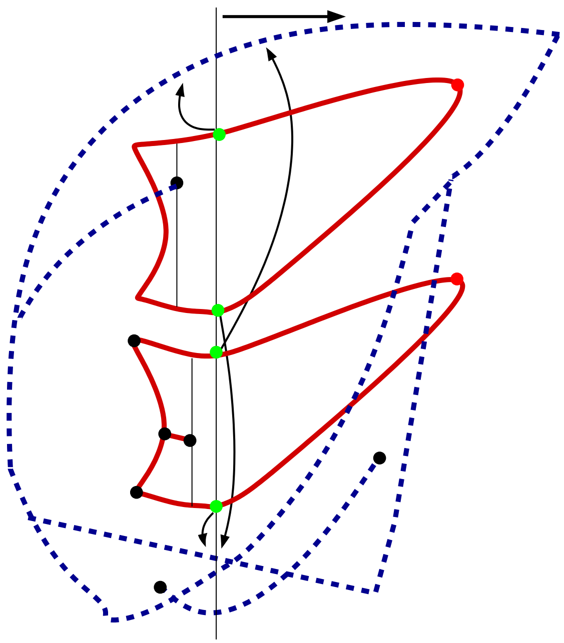
$n = r + b + k$

- Nie mehr als $O(r + b + k)$ Punkte in ES
- Nie mehr als $O(r + b + k)$ Regionen in SSS
- Einfügen in ES: $O(\log(r + b + k))$
- Einfügen in SSS: $O(\log(r + b + k))$
- Nicht mehr als $O(r + b + k)$ Ereignisse:
  - Rot/Rot, Blau/Blau: r und b
  - Rot/Blau, neu aber in $O(r + b + k)$
  - Neue Punkte: max. $k + r + b$

Insgesamt: $O(n \log n)$

Analyse: Red-Blue Merge Th. 2.22

- Problem (Besonderheiten):
- Eine bewachte Kante gehört zu vielen Zellen!
- Nicht für alle Zellen prüfen!!
- Abhilfe: In Listen zusammenfassen!
- Nur mit oberen/unteren den Schnitt testen!



Initialisierung Red-Blue Merge Th. 2.22

- Weitere Besonderheiten
- Initialisierung
- Zu Beginn nur zwei einzelne Bögen
- Natürliche Begrenzung (häufig)
- Unendliche Zellen
- Beispiel: Tafel!

Zellenberechnung: Th. 2.23

n X-monotone Kurvenstücke von denen sich zwei nur s mal schneiden, x gegeben. Zelle Z_x kann in Zeit $O(\lambda_{s+2}(n))\log^2 n)$ berechnet werden.

Fahrplan!!

- Divide and Conquer!!
- Teile Segmente in zwei gleichgroße Mengen Z_1, Z_2
- Berechne Z_{1x} und Z_{2x}
- Merge zu $\{Z_1 \cup Z_2\}_x$
- Spezieller Merge wegen Schnitt mit x
- RED BLUE Merge
- Merge: Komplexität des Ergebnisses

Beweis: Th. 2.22

Divide and Conquer

$$
\begin{aligned}
T(n) &\leq 2T\left(\frac{n}{2}\right) \text{(Rek.)} + C \times \lambda_{s+2}(n)\log(\lambda_{s+2}(n)) \text{(R/B-Merge)}\\
&\leq 2\left(2T\left(\frac{n}{4}\right) + C\lambda_{s+2}\left(\frac{n}{2}\right)\log\frac{n}{2}\right) + C \times \lambda_{s+2}(n)\log n\\
&\vdots \quad \vdots\\
&\leq (n)\,(T(1)+C) + C\sum_{i=0}^{\log n}\left(\lambda_{s+2}(n)\log\frac{n}{2^i}\right)\\
&\in O(\lambda_{s+2}(n)\log^2 n)
\end{aligned}
$$

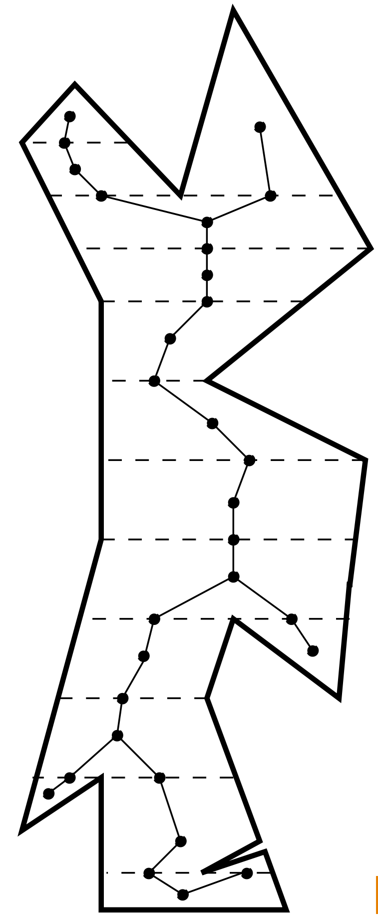
Anwendungen: Kap. 2.2.2

- Polygonaler Roboter R mit $|R| = m$
- Polygonale Szene n Ecken
- Reine Translationsbewegung
- Startposition s, Endposition t
- Kollisionsfreie Bahn von s nach t

Alg. 2.7

Preprocessing:

- Arrang. $2mn$ Linienseg.(Ecke/Kante)
- Ber. Zelle Z, die s enthält: Komplexität $O(\lambda_{(1+2)}(mn))$, Laufzeit $O(\lambda_{(1+2)}(mn)\log^2(mn))$
- Trapezzerlegung, Zusammenhangsgraph: Seidel $O(\lambda_{(1+2)}(mn)\log^*(mn))$ (Sweep)



Query: gegebenes t:

- Trapez, das t enthält: $O(\log(mn))$
- Pfad s nach t im Zusammenhangsgraph: $O(\lambda_{(1+2)}(mn))$

Theorem 2.24

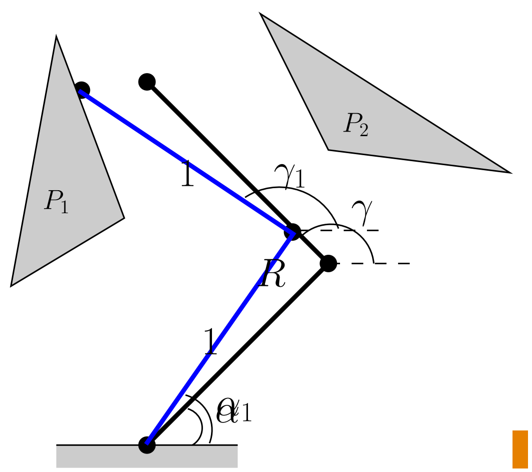
Translation von R polygonaler Roboter mit m Ecken, in einer Umgebung mit polygonalen Hindernissen P_i mit insgesamt n Ecken.

Gegeben seien Start– und Zielposition s, t.

Dann kann in Zeit $O(mn\,\alpha(mn)\log^2(mn))$ eine kollisionsfreie Translation von s nach t bestimmt werden oder festgestellt werden, dass keine solche existiert.

Anwendungen: Kap. 2.2.3

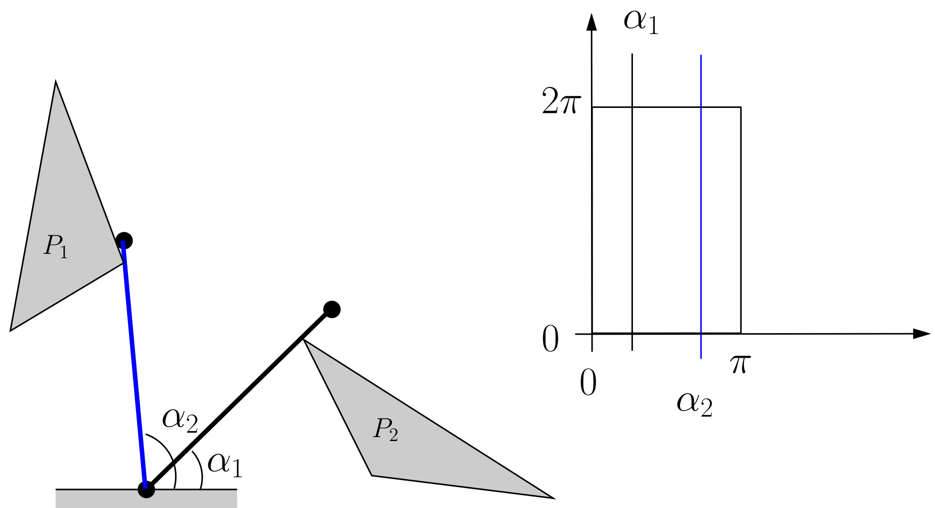
- Allgemeinheit der Konstruktion ausnutzen
- n Bögen begrenzen Konfigurationsraum
- Beispiel: Podest, Roboterarm mit zwei Gelenken
- Zwei Freiheitsgerad: Tupel des Konfigurationsraumes!!
- Hindernisse, normierte Armlänge



Wodurch wird eine Zelle begrenzt? 1.

Einschränkung unterer Bogen: α_1, α_2

Zwei Kanten im Konfigurationsraum!!

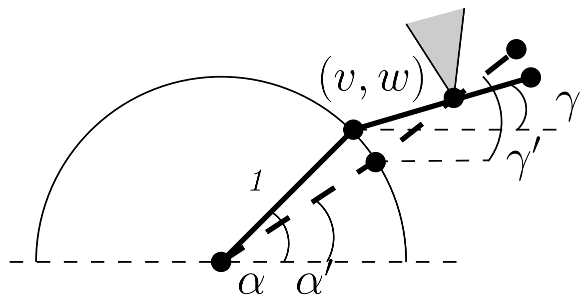


Wodurch wird eine Zelle begrenzt? 2.

Kontakt: Hindernisecke mit oberem Arm! Entlangschieben!

Kurve im Konfigurationsraum!!

Geschickte Parametrisierung wählen! (Tafel)

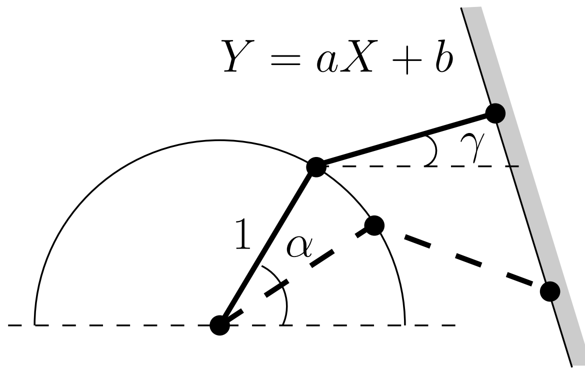


Wodurch wird eine Zelle begrenzt? 3.

Kontakt: Roboterecke mit Hinderniskante! Entlangschieben!

Kurve im Konfigurationsraum!!

Geschickte Parametrisierung wählen! (Tafel)



Algebraische Kurven!

1. 2 Geraden $x = \cos(\alpha_i)$ $i = 1, 2$
2. n Kurven: (v, w) fest! $\{(x, y) | (2wy^2)^2(1 - x^2) = (v^2 - 2xv + x^2 - y^2v^2 + 2xvy^2 - w^2y^2 - y^2)^2\}$
3. n Kurven: (a, b) fest! $\{(x, y) | ((a(x + y) + b)^2 - 2 + x^2 + y^2)^2 = (1 - x^2)(1 - y^2)\}$

Multivariante Polynome vom Grad ≤ 6!

Theorie: Je zwei maximal 6^2 Schnitte! Numerisch berechnen!

Th. 2.22 anwenden: Bahnplanung in $O(\lambda_{(36+2)}(n) \log^2 n)$!!